SOCIAL SENTIMENT ANALYSIS

ALAISSA SHAIKH

Social Sentiment Analysis

Also known as **OPINION MINING**

Social sentiment analysis is the interaction of the people's attitude and feeling towards a specific brand, service or products.

It is a method of computational identifying and categorizing opinions.

Types of Social Sentiment Analysis

There are types of Social Sentiment Analysis based on the Polarity of text:

1. Positive
2. Negative
3. Neutral

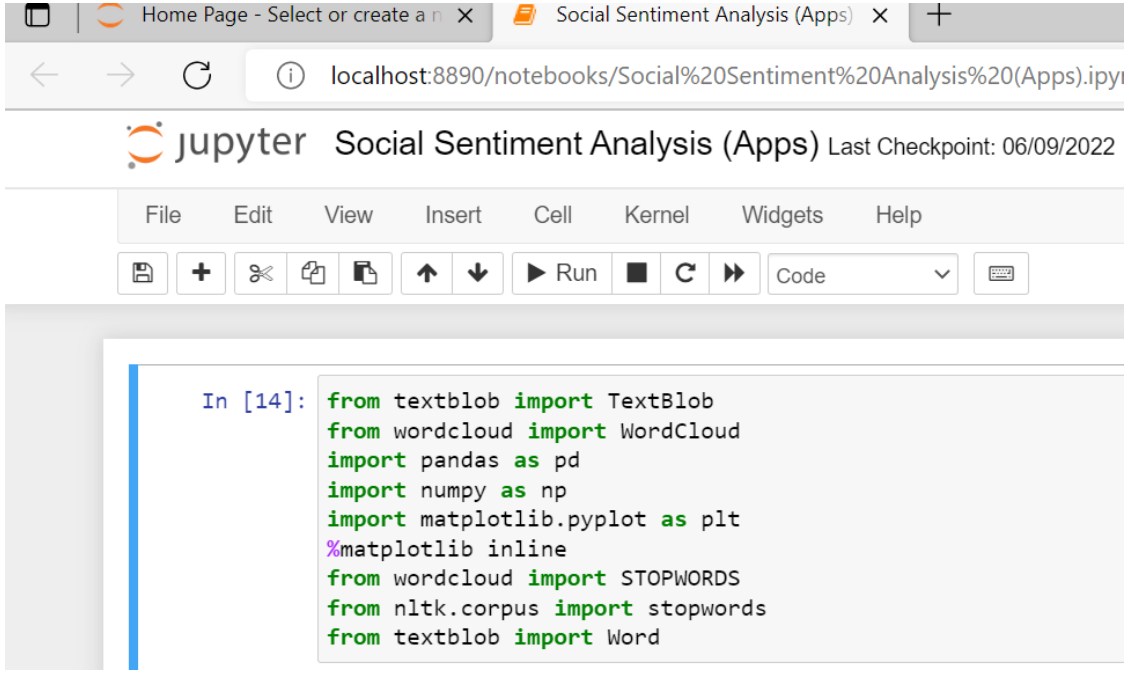
Purpose of Social Sentiment Analysis

1. Helps business to determine the demand in the market.
2. Analyze the customer requirements .
3. Stores data in efficient, cost-friendly manner.
4. Exploration of Subjective Opinion.

Brief Details of Social Sentiment Analysis

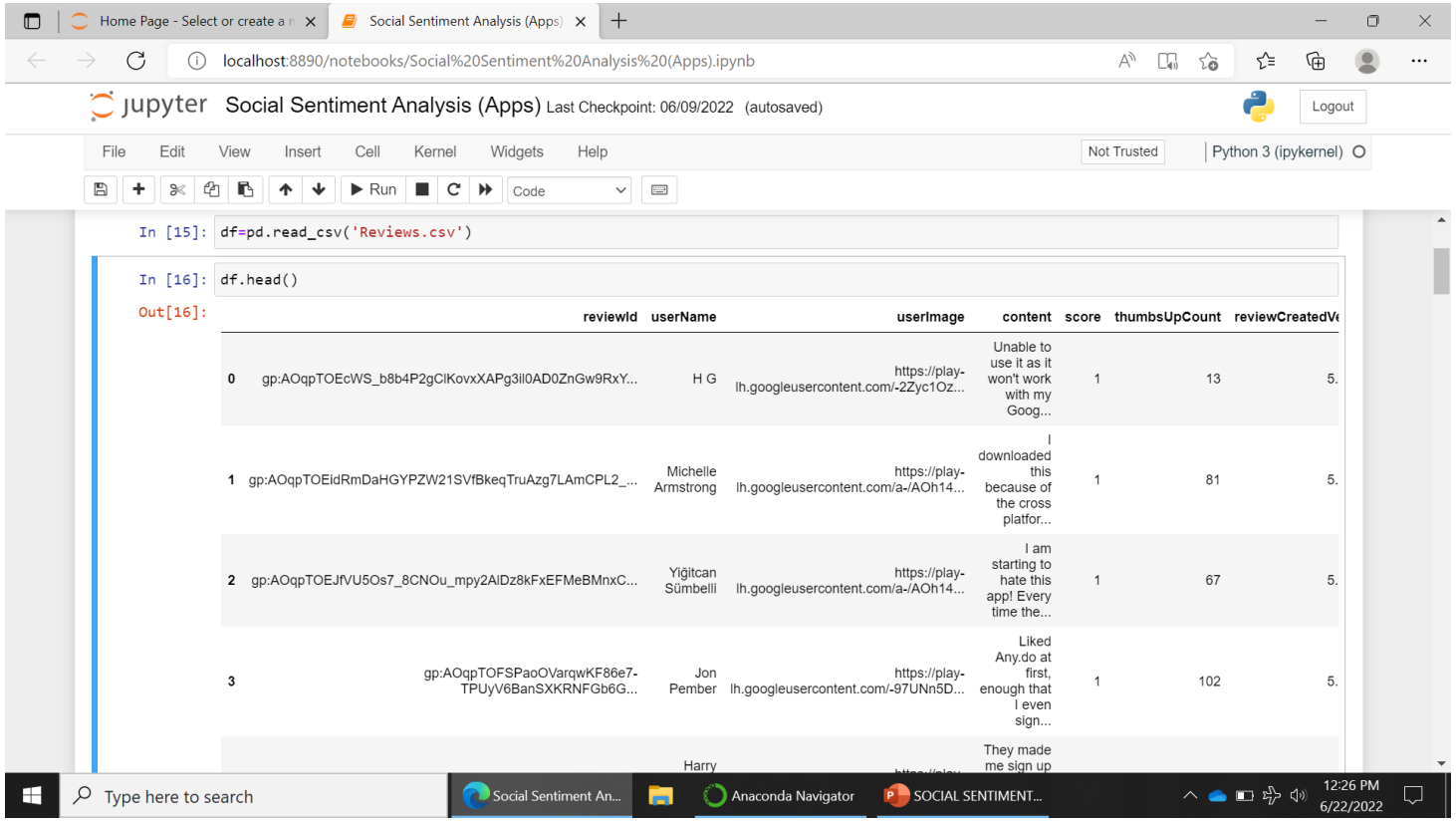
Importing the required libraries:

1. Textblob – for processing text-based data
2. Word cloud – for generating word cloud
3. Pandas – for data cleaning and data processing
4. Numpy – for mathematical functions
5. Matplotlib – for visualization
6. Stopwords – for filteration



Importing Dataset using pd.read_csv:

used to read a CSV file into a dataframe




```
In [17]: df.shape
Out[17]: (16388, 12)
```

```
In [18]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16388 entries, 0 to 16387
Data columns (total 12 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   reviewId              16388 non-null  object
 1   userName              16388 non-null  object
 2   userImage             16388 non-null  object
 3   content               16388 non-null  object
 4   score                 16388 non-null  int64
 5   thumbsUpCount         16388 non-null  int64
 6   reviewCreatedVersion  14028 non-null  object
 7   at                    16388 non-null  object
 8   replyContent          8488 non-null   object
 9   repliedAt             8488 non-null   object
 10  sortOrder             16388 non-null  object
 11  appId                 16388 non-null  object
dtypes: int64(2), object(10)
memory usage: 1.5+ MB
```

```
In [19]: df.content.head()

Out[19]: 0    Unable to use it as it won't work with my Goog...
         1    I downloaded this because of the cross platfor...
         2    I am starting to hate this app! Every time the...
         3    Liked Any.do at first, enough that I even sign...
         4    They made me sign up for premium just to have ...
         Name: content, dtype: object

In [22]: df['content']=df['content'].apply(lambda x: " ".join(x.lower() for x in x.split ())).str.replace('[^\w\s]'," ")
         df.content.head()

         C:\Users\ashaikh\AppData\Local\Temp\ipykernel_4804\792725223.py:1: FutureWarning: The default value of regex will change from T
         rue to False in a future version.
           df['content']=df['content'].apply(lambda x: " ".join(x.lower() for x in x.split ())).str.replace('[^\w\s]'," ")

Out[22]: 0    unable to use it as it won t work with my goog...
         1    i downloaded this because of the cross platfor...
         2    i am starting to hate this app every time they...
         3    liked any do at first enough that i even signe...
         4    they made me sign up for premium just to have ...
         Name: content, dtype: object

In [23]: df.columns
```

For removing whitespace characters and converting the data set into lowercase for easier data preprocessing.

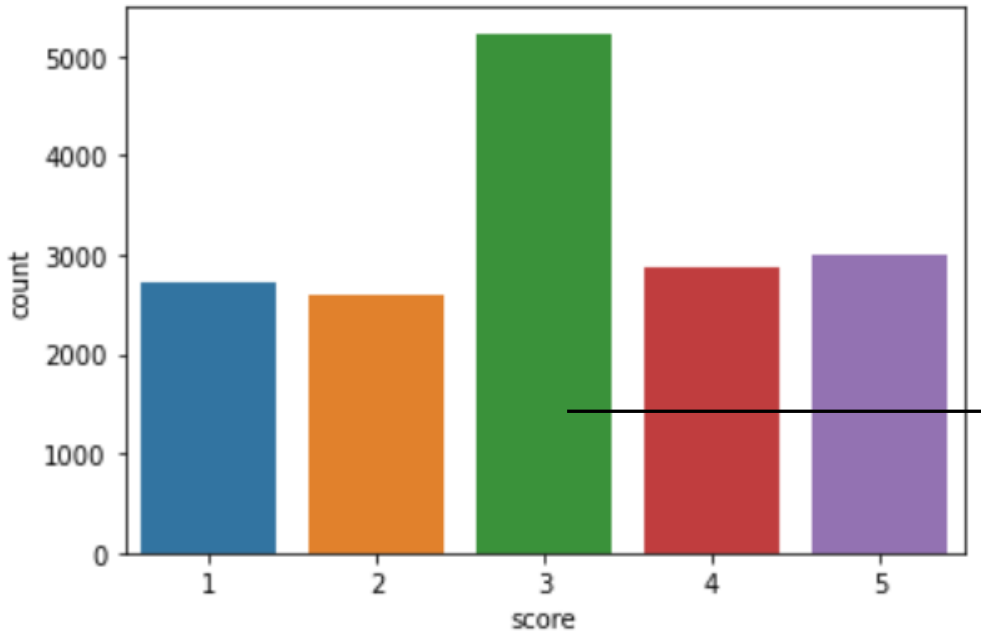
To review all the columns, present in the dataset

To show the polarity of the scores present in the dataset

Where:
3 is the neutral polarity
1, 2 are the negative polarity
4,5 are the positive polarity

```
In [23]: df.columns

Out[23]: Index(['reviewId', 'userName', 'userImage', 'content', 'score',
                'thumbsUpCount', 'reviewCreatedVersion', 'at', 'replyContent',
                'repliedAt', 'sortOrder', 'appId'],
               dtype='object')

In [24]: df.score.value_counts()

Out[24]: 3    5226
         5    3000
         4    2860
         1    2716
         2    2586
         Name: score, dtype: int64
```

```
In [25]: import seaborn as sns

In [26]: sns.countplot(data=df, x='score')

Out[26]: <AxesSubplot:xlabel='score', ylabel='count'>
```



Seaborn library is used for statistical plotting.

Here, in the given dataset we have plotted a bar graph of score vs count

Where, we can see the neutral polarity is high.

Data frame, exploratory data analysis

In [27]:
```
reviews=df
reviews.dropna(inplace=True)
```

In [28]:
```
score_1 = reviews[reviews['score'] == 1]
score_2 = reviews[reviews['score'] == 2]
score_3 = reviews[reviews['score'] == 3]
score_4 = reviews[reviews['score'] == 4]
score_5 = reviews[reviews['score'] == 5]
```

In [31]:
```
reviews_sample = pd.concat([score_1,score_2,score_3,score_4,score_5],axis=0)
reviews_sample.reset_index(drop=True,inplace=True)
```

Storing values in Single String

Storing values in Multiple String

Forming the WordCloud

NEGATIVE REVIEWS

```
In [33]: negative_reviews = reviews_sample[reviews_sample['score'].isin([1,2])]
         positive_reviews = reviews_sample[reviews_sample['score'].isin([1,2])]
         negative_reviews_str = negative_reviews.content.str.cat()
         positive_reviews_str = positive_reviews.content.str.cat()
```

```
In [34]: wordcloud_negative = WordCloud(background_color='white').generate(negative_reviews_str)
         wordcloud_positive = WordCloud(background_color='white').generate(positive_reviews_str)
```

```
In [35]: fig = plt.figure(figsize=(10,10))
         ax1 = fig.add_subplot(211)
         ax1.imshow(wordcloud_negative,interpolation='bilinear')
         ax1.axis("off")
         ax1.set_title("NEGATIVE REVIEWS", fontsize = 20)
```

```
Out[35]: Text(0.5, 1.0, 'NEGATIVE REVIEWS')
```

POSITIVE REVIEWS

```
In [36]: fig = plt.figure(figsize=(10,10))
         ax1 = fig.add_subplot(211)
         ax1.imshow(wordcloud_positive,interpolation='bilinear')
         ax1.axis("off")
         ax1.set_title("POSITIVE REVIEWS", fontsize = 20)
```

```
Out[36]: Text(0.5, 1.0, 'POSITIVE REVIEWS')
```

In [39]:
```
import re
import os
import sys
import ast
plt.style.use('fivethirtyeight')
cp = sns.color_palette()
from vaderSentiment.vaderSentiment import SentimentIntensityAnalyzer
analyzer = SentimentIntensityAnalyzer()
```

In [40]:
```
emptyline=[]
for row in df['content']:
    vs=analyzer.polarity_scores(row)
    emptyline.append(vs)
df.sentiments=pd.DataFrame(emptyline)
df.sentiments.head()
```

Out[40]:

	neg	neu	pos	compound
0	0.040	0.874	0.086	0.7264
1	0.000	0.974	0.026	0.2846
2	0.172	0.807	0.021	-0.9212
3	0.051	0.861	0.087	0.5574
4	0.000	0.969	0.031	0.0772

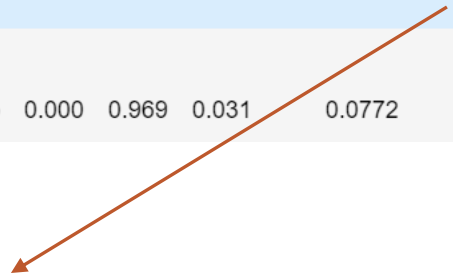
Analyzing the polarity of the text.

Finding the compound value with the help of negative, neutral and positive reviews

In [42]:
```
df_c['Sentiments'] = np.where(df_c['compound'] >= 0 , 'Positive','Negative')
df_c.head()
```

Out[42]:

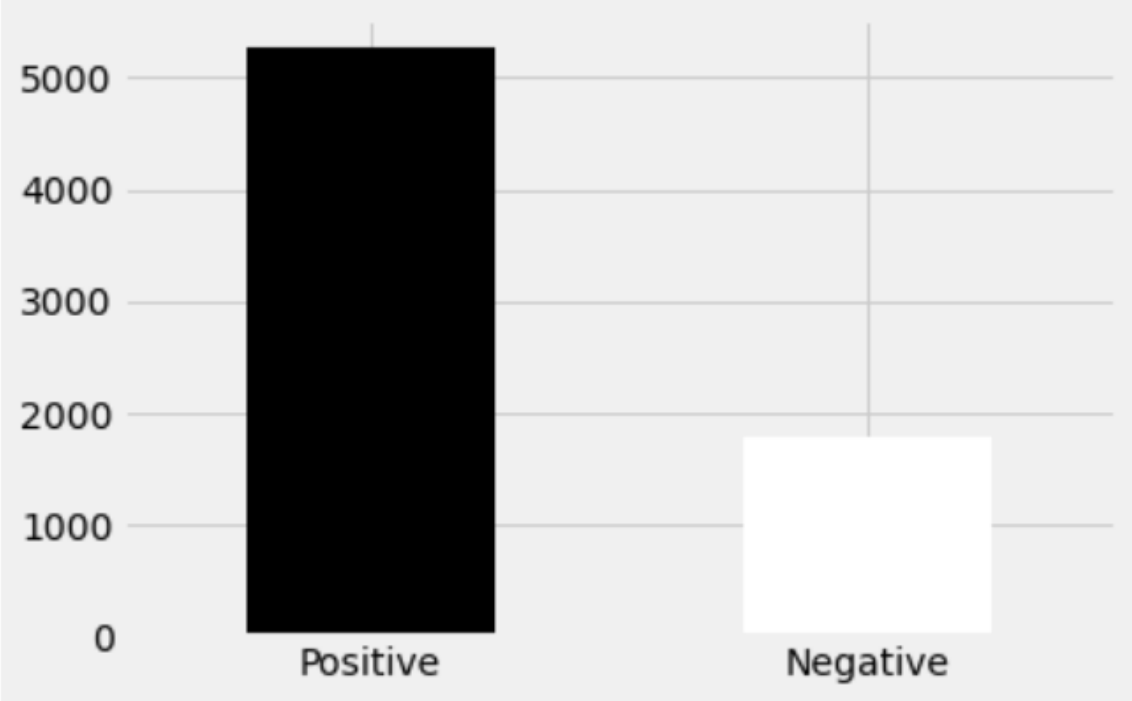
content	score	thumbsUpCount	reviewCreatedVersion	at	replyContent	repliedAt	sortOrder	appId	neg	neu	pos	compound	Sentiments
unable to use it as it won t work with my goog...	1	13	5.7.0.20	2021-01-11 01:12:58	As our team explained profusely, we sync data ...	2021-01-10 12:58:09	most_relevant	com.anydo	0.040	0.874	0.086	0.7264	Positive
i ownloaded this because of the cross platfor...	1	81	5.7.0.20	2020-12-22 18:31:11	The Premium ad only shows up when first openin...	2020-12-23 19:58:46	most_relevant	com.anydo	0.000	0.974	0.026	0.2846	Positive
i am starting to hate this app every time they...	1	67	5.7.0.10	2020-12-02 15:52:24	Hi, please note that these issues usually are ...	2020-12-03 20:47:02	most_relevant	com.anydo	0.172	0.807	0.021	-0.9212	Negative
ked any do at first enough that i even signe...	1	102	5.7.0.10	2020-11-22 23:42:56	Please note that Any.do integrates directly wi...	2020-11-25 13:30:01	most_relevant	com.anydo	0.051	0.861	0.087	0.5574	Positive
they made me sign up or premium ust to have	1	2	5.7.0.20	2020-12-31 01:49:52	We never require users to upgrade,	2020-12-31 20:40:53	most_relevant	com.anydo	0.000	0.969	0.031	0.0772	Positive



Reviewing the compound and adding the Sentiments

```
In [46]: result=df_c['Sentiments'].value_counts()
         result.plot(kind="bar", rot=0, color=['black','white'])
```

```
Out[46]: <AxesSubplot:>
```



Plotting the graph

Here, the graph shows the comparison of positive and negative reviews.

Which concludes that the positive reviews are more than the negative ones

CONCLUSION:

Social Sentiment Analysis keeps a track of the mentions.

It analyze the sentiments and information.

It monitors reviews and give a deeper analysis.

Helps finding the important attributes.

It spots opportunities and helps in improvements.

Helps borden the scope of search.